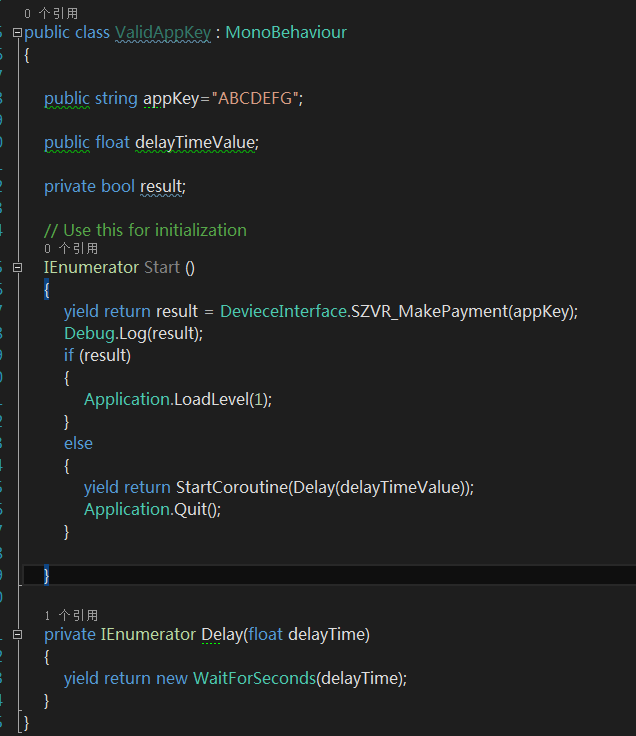
ThreeGlasses接口文档及详细使用说明

# DeviceInterface类

1. DeviceInterface.Initialize ()；设备初始换
2. DeviceInterface.GetIPD ()；获取瞳孔之间的距离
3. DeviceInterface.UpdateDeviceTesting ()；检测设备状态
4. DeviceInterface.DeleteDevice ()；移除设备
5. DeviceInterface.GetCameraOrientation()；获取头戴设备的方向
6. DeviceInterface.ResetOrientation()；重置正方向，以当前头部旋转位置为正方向
7. DeviceInterface.SZVR\_MakePayment(string appKeyValue);收费扣除接口



# SZVRDevice类

1. SZVRDevice device = gameObject.AddComponent<SZVRDevice>();脚本首先绑定在场景内主摄像机上，实现曲面变形后的左右分屏效果。



1. SZVRDevice.direction

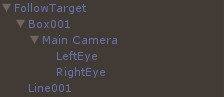
这个静态变量是当前传感器方向数据+当前角色的朝向，如果FollowTarget != null，那么这个参数会加上followTarget.rotation。

1. Near Clip Plane & Far Clip Plane

左右两只眼睛的相机的Clip Plane，来调节可视距离。

1. FollowTarget

此变量是一个Transform类型，主要能实现摄像机目标跟随的效果。



如上图Box001为带轨迹动画的物体，那么我们将主摄像机拖放到Box001下，将目标相机对准跟随物体，然后将Box001拖动到FollowTarget这个变量中就能实现摄像机跟随

Box001的效果。

1. Eye To Neck Height

眼睛到脖子的高度，将主摄像机放置到脖子的部位，设置此参数可以模仿眼睛的具体位置，此参数和实际的比例相同（例如：0.02 = 2mm）。

# SZVRCanvas类

1. SZVRCanvas canvas = gameObject.AddComponent<SZVRCanvas>();脚本首先绑定在场景内主摄像机上，实现曲面变形后的左右分屏效果



1. canvas.enabled = true;

脚本首先绑定在场景内主摄像机上，实现曲面变形后的左右分屏效果，反之为正常画面。

# FAQ

1、是否能控制sensor数据的使用和关闭？可以设置一个bool的变量来控制是否设置物体的旋转角度，如下：

bool isController = false; Quaternion q = Quaternion.identity;

void Start() {

q = transform.rotation;

}

void Update()

{

if(isController)

{

q = SZVRDevice.direction;

transform.rotation = q;

}

else

transform.rotation = q;

}